Cataloguing General Graphs by Point and Line Spectra

Stephen A. Fulling, Itshak Borosh, and Andrea da Conturbia*
Department of Mathematics
Texas A&M University
College Station, Texas, 77843, U.S.A.

July 6, 1998

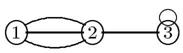
Abstract

Certain perturbative expansions in physics are sums over all multigraphs with loops — i.e., graph-like structures in which each point can be connected to any other point, including itself, by an arbitrary number of lines. We treat the problem of exhibiting, or at least counting, all such objects for all reasonably small numbers of points and lines, as well as the associated problem of determining the number of labeled graphs corresponding to a given unlabeled one. Symbolic computation (specifically, *Mathematica*) has proven very useful here. It is helpful to classify the graphs further according to the distribution of the lines among the pairs of points (*line spectrum*) and simultaneously the distribution of the ends of the lines among the points (*point spectrum*). The basic tool is a generating function in which each coefficient is the number of general graphs with a given line spectrum and point spectrum.

1 Introduction

1.1 Terminology

We follow the conventions of Harary and Palmer (1973) for the most part. Informally, "graph" will mean "general graph" (or "multigraph with loops"). A graph in the usual, stricter sense will be called a "simple graph". We omit the proofs of the well known and fairly obvious propositions stated here.



$$\begin{array}{ccc} 0 & 3 & 0 \\ & 0 & 1 \\ & & 1 \end{array}$$

Figure 1: A labeled graph and the nonredundant part of its adjacency matrix.

Definition 1 A *labeled general graph on p points* is an assignment of a nonnegative integer to each unordered pair (with repetitions allowed) of positive integers $\le p$ (*points*). It is equivalent to an *adjacency matrix*, a symmetric $p \times p$ matrix of nonnegative integers. Pictorially, for $1 \le j, k \le p$, the (j, j) matrix element is the number of *loops* attached to the point labeled j, and the (j, k) element is the number of *links* connecting the points j and k. Loops and links collectively are *lines*. (See Figure 1.) □

*Present address: 26 Britten Rd., Green Village, Chatham Township, New Jersey, 07935, U.S.A.

A common alternative terminology to "point" and "line" is "vertex" and "edge". The former seems more appropriate to general graphs, where most lines must be drawn curved.

Definition 2 An *unlabeled (general) graph* is an equivalence class of labeled graphs under permutation of the points. (That is, the numbers 1 to p are replaced by indistinguishable abstract points.) □

Definition 3 The *multiplicity* of an unlabeled graph is the number of distinct labeled graphs that represent it. □

Definition 4 The *automorphism number*, $\mathrm{Aut}(G)$, of a labeled graph G is the order of its *automorphism group*; that is, the number of permutations of its points that leave the graph invariant. □

Proposition 1 $\mathrm{Aut}(G)$ *depends only on the unlabeled graph structure of* G. *The multiplicity of that graph is equal to*

$$\frac{p!}{\mathrm{Aut}(G)}\,,$$

where p *is the number of points.*

Definition 5 A *(p,q)-graph* is a (general) graph on p points with a total of q lines. We denote by r $(\leq q)$ the total number of links in such a graph. $\mathcal{G}_{pq}$ is the set of all (p,q)-graphs. □

Definition 6 A *simple graph* (usually called just "graph") is a general graph without loops in which each pair of points is joined by at most one line. That is, the only allowed adjacency matrix elements are 0 and 1, with all zeros on the diagonal. □

Definition 7 A graph is *connected* if every pair of points in it is connected by a chain of links. We denote the set of all connected (p,q)-graphs by $\mathcal{G}^{\mathrm{c}}_{pq}$. □

Proposition 2 *A simple* (p,q)*-graph must have* $q \leq p(p-1)/2$. *A connected* (p,q)*-graph must have* $q \geq r \geq p-1$.

1.2 An application

General graphs arise naturally in perturbation expansions in statistical mechanics and quantum mechanics. In recent years, T. A. Osborn and coworkers have made extensive use of them. We discuss here only the simplest such application, to the (time-dependent) Schrödinger equation with a time-independent scalar potential, with mass and time as expansion parameters (Fujiwara, Osborn, and Wilk, 1982). (More recent developments include electromagnetic vector potentials (Molzahn, 1992; Barvinsky, Osborn, and Gusev, 1995), expansion in Planck's constant (Molzahn and Osborn, 1994), and Heisenberg equations of motion (Osborn and Molzahn, 1995).)

Consider the initial-value problem for the Schrödinger equation

$$\begin{aligned} i\hbar\,\frac{\partial\Psi}{\partial t} &= -\,\frac{\hbar^2}{2m}\,\nabla^2\Psi + \lambda V(\mathbf{x})\Psi \\ &\equiv H\Psi \end{aligned} \tag{1}$$

where $V \in C^\infty(\mathbf{R}^d)$. The *propagator* $K(\mathbf{x},\mathbf{y};t;\hbar,m,\lambda)$ is the integral kernel of the operator $e^{-itH/\hbar}$. But

$$\begin{aligned} \frac{tH}{\hbar} &= -\,\frac{\hbar t}{2m}\,\nabla^2 + \frac{\lambda t}{\hbar}\,V \\ &\equiv -\tfrac{1}{2}A\nabla^2 + BV. \end{aligned} \tag{2}$$

Apart from $\mathbf{x}$ and $\mathbf{y}$, therefore, the only independent variables in the problem are

$$A \equiv \frac{\hbar t}{m} \quad \text{and} \quad B \equiv \frac{\lambda t}{\hbar}\,. \tag{3}$$

One investigates K perturbatively as some combination of A and B becomes small. Physically,

$B \to 0 \quad (\lambda \to 0)$ is the coupling-constant expansion (an expansion in powers of V);

$A \to 0 \quad (m \to \infty)$ is the adiabatic expansion (an expansion in derivatives of V);

$A, B \to 0 \quad (t \to 0)$ is the short-time expansion;

$A/B \to 0 \quad (\hbar \to 0)$ is the semiclassical (WKB) expansion.

For more general operators H (such as Laplace–Beltrami operators on compact manifolds) the calculation of small-time expansions is a major industry (e.g., Gilkey (1975), Avramidi (1991)). There the terms are known only recursively, and there is no visible pattern in their numerical coefficients. Indeed, even specifying a linearly independent list of the terms that may appear at each order is a nontrivial exercise in group representation theory (Fulling, King, Wybourne, and Cummins, 1992). It is therefore significant that *for the problem (1) the series can be found in closed form, and the coefficients have a relatively simple combinatorial structure.*

To state the graph formula of Osborn et al. we need one more notational convention: Let α label the $\frac{1}{2}p(p+1)$ nonredundant places in the adjacency matrix (i.e., the unordered pairs (i_α, j_α) of points, not necessarily distinct); let l_α be the corresponding matrix element (i.e., the number of lines connecting those two points in the particular graph at hand). Also, let n be the total number of loops (the trace of the matrix).

Theorem 1 (Fujiwara et al., 1982; Molzahn and Osborn, 1987; Osborn and Molzahn, 1991) *The integral kernel of the operator $e^{-itH/\hbar}$, H defined in (1) or (2), is*

$$K(\mathbf{x},\mathbf{y};t) = \left(\frac{m}{2\pi i\hbar t}\right)^{\frac{d}{2}} e^{im|\mathbf{x}-\mathbf{y}|^2/2\hbar t}\, e^{L},$$

where L has the asymptotic expansion

$$L \sim \sum_{p=1}^{\infty}\sum_{q=0}^{\infty} (-i)^{p+q} A^q B^p \sum_{G\in\mathcal{G}^c_{pq}} L_{pq}[G];$$

$$L_{pq}[G] \equiv \left[2^n p! \prod_{\alpha=1}^{\frac{1}{2}p(p+1)} l_\alpha!\right]^{-1} \int_{[0,1]^p} d^p\xi \prod_{\alpha} \left[g(\xi_{i_\alpha},\xi_{j_\alpha})\nabla_{i_\alpha}\cdot\nabla_{j_\alpha}\right]^{l_\alpha} \prod_{i=1}^{p} V(\mathbf{z}_i),$$

where

$$\mathbf{z}_i \equiv \mathbf{y} + \xi_i(\mathbf{x}-\mathbf{y})$$

(the parametrized "free" path from $\mathbf{y}$ to $\mathbf{x}$) and

$$g(\xi_i,\xi_j) \equiv \xi_<(\xi_> - 1)$$

(the Green function arising in linearization of the classical equation of motion about the free path).

That is, the terms are represented by connected graphs, in which vertices stand for factors V, and lines stand for pairs of derivatives, "contracted" in the sense of tensor calculus.

Expansion of the exponential produces a series of the same structure, except that the graphs are no longer required to be connected:

Corollary 1 *In the situation of Theorem 1 one has*

$$e^L \sim 1 + \sum_{p=1}^{\infty} \sum_{q=0}^{\infty} (-i)^{p+q} A^q B^p \sum_{G \in \mathcal{G}_{pq}} L_{pq}[G].$$

For example, let us look at all the terms in Corollary 1 with $p = 2$, $q = 1$. The single-loop graph G_1 with adjacency matrix $\begin{smallmatrix}1 & 0\\ & 0\end{smallmatrix}$ yields

$$L_{21}[G_1] = \frac{1}{4} \int_0^1 d\xi_1 \int_0^1 d\xi_2 \, \xi_1(\xi_1 - 1) \, \nabla^2 V(\mathbf{z}_1) \, V(\mathbf{z}_2).$$

The connected graph G_2 with matrix $\begin{smallmatrix}0 & 1\\ & 0\end{smallmatrix}$ gives

$$L_{21}[G_2] = \frac{1}{2} \int_0^1 d\xi_1 \int_0^1 d\xi_2 \, \xi_<(\xi_> - 1) \, \nabla V(\mathbf{z}_1) \cdot \nabla V(\mathbf{z}_2).$$

Finally, $G_3 = \begin{smallmatrix}0 & 0\\ & 1\end{smallmatrix}$ just gives another copy of $L_{21}[G_1]$. If $\mathbf{y} = \mathbf{x}$, the integrals can be evaluated:

$$2L_{21}[G_1] + L_{21}[G_2] = -\frac{1}{12} V \nabla^2 V - \frac{1}{24} |\nabla V|^2.$$

Theorem 1 as stated does not yield a true semiclassical expansion (one in A/B). However, the same graphical analysis can be applied in the construction of a semiclassical expansion, when the free paths are replaced by exact solutions of the classical equation of motion (Molzahn and Osborn, 1994; Osborn and Molzahn, 1995).

1.3 The problem

The redundancy of G_1 and G_3 in the foregoing example demonstrates a shortcoming of the asymptotic expansion formulated in Theorem 1 and Corollary 1: Isomorphic labeled graphs give identical contributions to the series. It would be better to sum over *unlabeled* graphs so as to get *linearly independent* terms. Proposition 1 implies:

Corollary 2 *The formulas in Theorem 1 and Corollary 1 remain valid when labeled graphs are replaced by unlabeled graphs and the factor $p!$ in the denominator is replaced by $Aut(G)$.*

Therefore, we will have *a complete, explicit solution of the propagator expansion problem* if we can, order by order, (1) explicitly list all the unlabeled graphs, and (2) calculate the automorphism number of each graph. This is the task to which the present paper is devoted.

One needs to study the following family of problems.

1 **Isomorphism:** Given two labeled (general) graphs, determine whether they are isomorphic (i.e., represent the same unlabeled graph).

2a **Automorphism:** Given a labeled graph, determine how many permutations of its vertices leave it invariant.

2b **Multiplicity:** Given an unlabeled graph, determine how many labeled graphs represent it.

(Problems 2a and 2b are equivalent, since the product of the two numbers is $p!$.) The next three problems assume that some *canonical ordering* of the labeled graphs has been defined.

3 **Normal Form:** Given a labeled graph, find the first labeled graph isomorphic to it. (This is to be taken as the canonical representation of the corresponding unlabeled graph. Thereby the unlabeled graphs inherit a canonical ordering.)

4 **Enumeration:** Count the unlabeled (p, q)-graphs.

5 **Catalogue:** Construct a list of all the unlabeled (p, q)-graphs in the canonical order.

It is fairly easy to see how each of these problems could be solved by brute force. However, the most obvious methods become inefficient for large values of p and q. The isomorphism problem for simple graphs has been much studied (Read and Corneil, 1977; Kučera, 1990, especially Sec. 9.4). It is conjectured (but not known) to be NP-complete. Read (1978) developed the procedure of "orderly generation" for cataloguing graphs and other combinatorial structures without subjecting each new candidate to an isomorphism check against all those previously generated. Further refinements were made by Goldberg (1993) (who showed that the enumeration of (simple) graphs can be done by an algorithm with "polynomial delay") and others (see Benecke et al. (1995) and references therein). Kerber and coworkers (Benecke et al., 1995; Wieland et al., 1996; Kerber et al., 1996) have developed software for solving molecular structure problems in chemistry which employs these methods and exploits special features of the chemical application to achieve additional speedups.

The present paper is not directed at these issues of computational complexity. The lack of a polynomial-time algorithm is probably not a serious concern in the context of the propagator-expansion problem. There the listing of the graphs is only a preliminary to the larger task of evaluating the terms they represent. In practice, therefore, one is likely to be interested only in rather modest values of p and q, say 5 to 10 at most, for which all the associated calculations are too large to be handled easily by human inspection but are not beyond the reach of computers.

Here we present a method of solving the enumeration problem by generating functions, following the general strategy of Pólya (Pólya and Read, 1987; Harary and Palmer, 1973; Parthasarathy, 1968). This provides a count of the unlabeled (p, q)-multigraphs with a given point and line spectrum (defined below), without the need to perform any isomorphism or canonicity checks. The classification of graphs thus obtained is sufficiently fine (i.e., the number of graphs in each bin is sufficiently small) that at the relatively small values of p and q here considered, there is no difficulty in subsequently cataloguing all the graphs by inspection.

2 Point and Line Spectra

This section is the core of the paper, in which, among other things, an algorithm is presented and implemented to find the number of (p, q)-graphs with a given arrangement of lines. (The precise meaning of "arrangement" will be formulated forthwith.)

2.1 Definitions

As previously remarked, a first step toward classifying graphs up to isomorphism is to count the number of lines attached to each point. The list of such numbers for a given graph is usually called the graph's *degree spectrum*. We call it *point spectrum*, because for multigraphs there is a useful line-oriented spectrum from which it needs to be distinguished.

Definition 8 The *point spectrum* of a general (p, q)-graph is the distribution of the ends of the lines among the points. It is represented by *a partition of* $2q$ *to maximum depth* p. $\square$

Definition 9 The *line spectrum* of a general (p, q)-graph is the distribution of the lines among their possible positions. It is represented by *a partition of* q *to maximum depth* $\frac{1}{2}p(p+1)$ (formed by the entries in the nonredundant part of the adjacency matrix). $\square$

For example, let $p = 3$ and $q = 5$. Then $2q = 10$ and $\frac{1}{2}p(p+1) = 6$. Therefore, the possible point spectra are

$$\overline{10},\ 91,\ 82,\ 73,\ 64,\ 55,\ 811,\ 721,\ 631,\ 541,\ 622,\ 532,\ 442,\ 433,$$

and the possible line spectra are

$$5,\ 41,\ 32,\ 311,\ 221,\ 2111,\ 11111.$$

(To prevent ambiguity, a bar is placed over a partition part whose decimal notation requires more than one digit.) It is sometimes useful to pad the description of a partition with trailing zeros out to the full quota of q or $\frac{1}{2}p(p+1)$ numbers.

A graph is not uniquely determined by its point spectrum and line spectrum, but the two together provide a much finer classification than either of them alone. In fact, for realistically small values of p and q, it is rather rare for a given pair of spectra to characterize more than one unlabeled graph. For example, our investigations show that when $p = 3$ and $q = 5$, of the 98 categories implied by the combination of 14 point spectra and 7 line spectra listed above, one is occupied by 4 graphs, two by 3 graphs, four by 2 graphs, 35 by only one graph, and 56 by none. In contrast, there are (for example) 7 graphs with the point spectrum 433, and 12 graphs with the line spectrum 221.

The classification might be made still finer by further complicating the generating function so as to distinguish between loops and links, and among graphs with different numbers of connected components.

Knowledge of how many graphs correspond to a given pair of spectra is very helpful in generating an actual list of the graphs, whether by hand (as we have done) or possibly automatically.

2.2 Generating function

We shall enumerate the graphs by spectra through an extension of the algorithm of Parthasarathy (1968) as presented on pp. 88–90 of Harary and Palmer (1973). (In the following propositions, graphs are understood to be unlabeled.)

Theorem 2 *The generating function for general graphs on p points, enumerated simultaneously by point and line spectra, is*

$$N(x_1, \ldots, x_p) = \frac{1}{p!} \sum_{\gamma \in S_p} \Theta \sum_{f=\gamma f} \prod_{j,k} C_{f_{jk}} (x_j x_k)^{f_{jk}}, \tag{4}$$

where Θ *indicates that the exponents in the* x*-monomial are to be arranged in descending order* [e.g., $\Theta(x_2^3 + x_1 {x_2}^2) \equiv x_1^3 + {x_1}^2 x_2)$]. *Here the domain of the functions* $f_{jk} \equiv f(\{j,k\})$ $(1 \le j,k \le p)$ *is the set of unordered pairs of points (possibly equal), the range is the nonnegative integers, and the elements* γ *of the symmetric group* S_p *act on these functions in the obvious way induced by the permutations of the points.*

The coefficient (guaranteed to be a nonnegative integer) of the monomial

$$C_0^{n_0} C_1^{n_1} \cdots C_i^{n_i} \cdots {x_1}^{m_1} \cdots {x_p}^{m_p} \tag{5}$$

in (4) is the number of general graphs (multigraphs with loops allowed) on p points with the point spectrum

$$m_1 \ldots m_p$$

and the line spectrum

$$\ldots \underbrace{i \ldots i}_{n_i \text{ times}} \ldots \underbrace{1 \ldots 1}_{n_1 \text{ times}} .$$

Corollary 3 *The generating function for general graphs on p points, enumerated by line spectrum, is* $N(1,\ldots,1)$. *The generating function for general graphs on p points, enumerated by point spectrum, is similarly obtained by setting all* $C_i = 1$ *in* $N(x_1,\ldots,x_p)$.

Corollary 4 *The generating function for general graphs on p points, enumerated by number of lines, q, is obtained by setting all* $C_i = 1$ *in* $N(t,\ldots,t)$. *That is, the total number of* (p,q)*-graphs is the coefficient of* t^{2q} *in that formal power series.*

PROOF **1** *Theorem 2 is the same as that stated on p. 89 of Harary and Palmer (1973), except that the basic object there,*

$$W(f) \equiv \prod_{j,k} (x_j x_k)^{f_{jk}}$$

(where f is an adjacency matrix), must be modified in three ways:

1. *The indices j and k are allowed to be equal (to represent loops).*
2. f_{jk} *can be any nonnegative integer, not just 0 or 1 (since we deal with multigraphs).*
3. *We include a formal factor* $C_{f_{jk}}$ *to keep track of the locations of the lines.*

In more detail: We wish the generating function $N(x_1,\ldots,x_p)$ *to contain exactly one term for each unlabeled graph, with "weight" (5) and numerical coefficient* 1. *By Burnside's Lemma (Weighted Form) [see Harary and Palmer, Sec. 2.3], this formal sum is equal to the average over* S_p *of the sum of the weights of all labeled graphs that are fixed points of the group element in question (since an unlabeled graph is the same thing as an orbit of the action of* S_p *on the labeled graphs). And that is exactly what is constructed in the main formula of the theorem, (4).*

Example: The terms in $N(x_1,x_2,x_3)$ of homogeneous degree 4 in the x variables turn out to be

$$C_2 C_0^5 ({x_1}^4 + {x_1}^2 {x_2}^2) + C_1^2 C_0^4 ({x_1}^3 x_2 + {x_1}^2 {x_2}^2 + 2{x_1}^2 x_2 x_3). \tag{6}$$

Setting all the C variables equal to 1, we obtain

$${x_1}^4 + {x_1}^3 x_2 + 2{x_1}^2 {x_2}^2 + 2{x_1}^2 x_2 x_3,$$

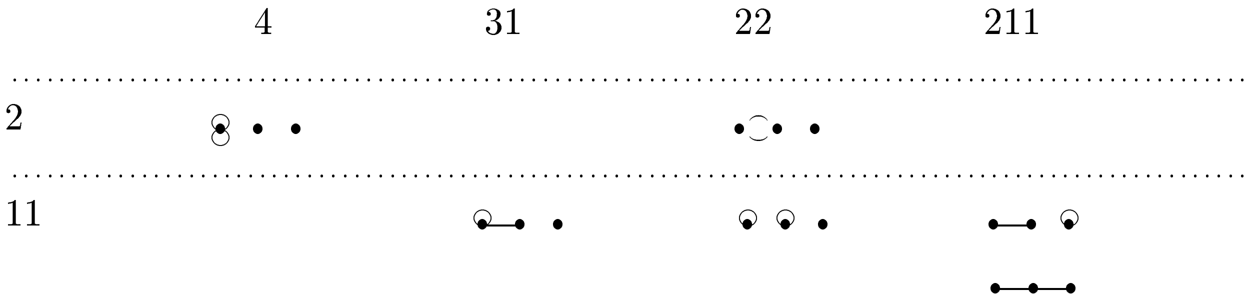


Figure 2: The classification of graphs with 3 points and 2 lines.

which predicts that there is one $(3,2)$-graph with point spectrum 400, one with spectrum 310, and two each with the spectra 220 and 211. On the other hand, setting all the x variables equal to 1 yields

$$2C_2{C_0}^5 + 4{C_1}^2{C_0}^4,$$

predicting two graphs with line spectrum 200000 and four graphs with line spectrum 110000. When all the variables are set to 1, the expression collapses to 6, which is the total number of general graphs with 3 points and 2 lines. We construct a table with rows labeled by the two possible line spectra and columns labeled by the four possible point spectra. Returning to (6), it is easy to fill in each entry in the table with the required graphs, producing Figure 2.

2.3 Algorithm

It remains to show how the series (4) can be constructed in practice. It is done by generalizing the example on p. 90 of Harary and Palmer (1973).

After application of the operator Θ, all group elements γ in the same conjugacy class of S_p make the same contribution to the sum. Therefore, one can just sum over the fixed points f of a canonical representative of each class and multiply by the cardinality of the class. The classes are well known to be in one-to-one correspondence with the partitions of p.

For example, in S_3 there are 2 cyclic permutations, 3 transpositions, and the identity permutation. A cyclic permutation mixes all the points among themselves, and all the point pairs among themselves. Hence the only allowable f are those in which all variables x_j appear equally often, and the full contribution of a cyclic to the sum is

$$\left({C_0}^3 + {C_1}^3{x_1}^2{x_2}^2{x_3}^2 + {C_2}^3{x_1}^4{x_2}^4{x_3}^4 + \cdots\right)^2, \tag{7}$$

where one factor is the contribution of the loops, the other that of the links. As the canonical transposition we choose the interchange of points 1 and 2; it leaves loops at point 3 and links between 1 and 2 alone, while it interchanges loops at 1 and 2 and the two kinds of links attached to 3. The series with these constraints is equivalent to the product

$$\begin{aligned}&\left(C_0 + C_1{x_3}^2 + C_2{x_3}^4 + \cdots\right)\left(C_0 + C_1x_1x_2 + C_2{x_1}^2{x_2}^2 + \cdots\right)\\ \times\ &\left(C_0^2 + C_1^2{x_1}^2{x_2}^2 + C_2^2{x_1}^4{x_2}^4 + \cdots\right)\left(C_0^2 + C_1^2x_1x_2{x_3}^2 + C_2^2{x_1}^2{x_2}^2{x_3}^4 + \cdots\right).\end{aligned} \tag{8}$$

The identity leaves all loops and links invariant, so its series contains all possible terms and can be constructed as the product of a simple series for each loop or link position:

$$\left(C_0 + C_1{x_1}^2 + C_2{x_1}^4\cdots\right)\left(C_0 + C_1{x_2}^2 + \cdots\right)\left(C_0 + C_1{x_3}^2 + \cdots\right)$$

$$\times \left(C_0 + C_1 x_1 x_2 + C_2 {x_1}^3 {x_2}^2 + \cdots\right)\left(C_0 + C_1 x_1 x_3 + \cdots\right)\left(C_0 + C_1 x_2 x_3 + \cdots\right). \tag{9}$$

Finally, these three pieces are combined as

$$N_3(x_1, x_2, x_3) = \tfrac{1}{6}\,\Theta[(9) + 3(8) + 2(7)]. \tag{10}$$

A *Mathematica* program to calculate (10) up through $q = 8$, along with the series in Corollaries 3 and 4, is

```
Share[]
orderRules = {
   x1^(n1_)*x2^(n2_) :> x1^n2*x2^n1 /; n2 > n1,
   x1^(n1_)*x3^(n3_) :> x1^n3*x3^n1 /; n3 > n1,
   x2^(n2_)*x3^(n3_) :> x2^n3*x3^n2 /; n3 > n2
   }

qmax = 8
max = 2(qmax+1)

link12 = Sum[t^(2j) c[j] x1^j x2^j, {j, 0, qmax}] +O[t]^max
link13 = link12 /. x2 -> x3
link23 = link13 /. x1 -> x2
point1 = Sum[t^(2j) c[j] x1^(2j), {j, 0, qmax}] + O[t]^max
point2 = point1 /. x1 -> x2
point3 = point1 /. x1 -> x3
pointpair = Sum[t^(4j) c[j]^2 x1^(2j) x2^(2j),
                                {j, 0, Floor[qmax/2]}] + O[t]^max
angle = Sum[t^(4j) c[j]^2 x1^j x2^j x3^(2j),
                                {j, 0, Floor[qmax/2]}] + O[t]^max
pointtriple = Sum[t^(6j) c[j]^3 x1^(2j) x2^(2j) x3^(2j),
                                {j, 0, Floor[qmax/3]}] + O[t]^max
linktriple = pointtriple

identity = link12*link13*link23*point1*point2*point3
transpos = link12*angle*pointpair*point3
cyclics = pointtriple*linktriple

pu3series = (1/6)(identity + 3*transpos + 2*cyclics)
pu3temp = (Expand[Normal[pu3series] * x1^2 x2^2 x3^2] //.
                                 orderRules) / (x1^2 x2^2 x3^2)
pu3 = Collect[pu3temp, Prepend[Table[c[qmax-i], {i,0,qmax}], t]]

pu3point = pu3 //. c[n_] -> 1
pu3line = pu3 //. {x1->1, x2->1, x3->1}
pu3total = pu3line //. c[n_] -> 1
```

```
Save["pu3.out", pu3, pu3point, pu3line, pu3total]
Quit[]
```

Three technical maneuvers were necessary here. First, *Mathematica* does not directly support power series in several variables (as opposed to power series whose coefficients are themselves power series), so it was necessary to replace each x_i by tx_i and use t as the formal expansion parameter. Second, the Θ operation is properly carried out by the substitution rules `orderRules` only if every variable appears with an explicit exponent. Therefore, one must temporarily multiply by $\prod x_i^2$ to eliminate any tacit exponents of 0 or 1. Finally, the `Collect` command groups and factors the terms so that a human reader can extract the desired information from the output.

Similar reasoning can be applied to any p. The decomposition of S_p into classes can be read off from Appendix III or formula (2.2.4) of Harary and Palmer. The contribution of the representative element of each class to the sum is a product of formal series like (8), consisting of one factor for each orbit of the group's action on the set of line positions. The action on the loops is just that on the points themselves, while the action on the links requires a bit more thought. The greatest subtlety concerns the links that join points belonging to two disjoint cycles in the cycle decomposition of a permutation; the result depends significantly on the greatest common divisor of the lengths of the two cycles (cf. pp. 86–87 of Harary and Palmer).

In this way we calculated the generating functions up to the order $p+q = 11$. It sufficed to write programs for $p \leq 7$, since, as will be clear from the theorems in Sec. 2.6, the enumerations for small q and large p follow easily from those for smaller p. The programs are available in Fulling (1995).

On a Sun–3/60 with 8 MB RAM and 32 MB swap space, the programs for $p \leq 3$ executed quickly, but those for $4 \leq p \leq 7$ required several hours each; attempting to run two programs at once caused an "Out of memory" error. More recently, all of the programs have been reexecuted on a Sun SPARCstation 10 (50 MHz, 368 MB) with a typical clock time of 5 minutes for the large programs, and a sampling of them have been reexecuted on a 486DX PC (66 MHz, 12 MB) with a typical clock time of 20 minutes.

2.4 Labeled graphs, multiplicities, and automorphisms

Counting labeled objects is easier than counting unlabeled objects, although there are more of them. The generating function for labeled graphs is just the identity part of the generating function described in Theorem 2 (without the factor $1/p!$). Therefore, it can be computed in the example $p = 3$ by replacing the last steps in the program presented in Sec. 2.3 by

```
pl3temp = (Expand[Normal[identity] * x1^2 x2^2 x3^2] //.
                                  orderRules) / (x1^2 x2^2 x3^2)
pl3 = Collect[pl3temp, Prepend[Table[c[qmax-i], {i,0,qmax}], t]]

pl3point = pl3 //. c[n_] -> 1
pl3line = pl3 //. {x1->1, x2->1, x3->1}
pl3total = pl3line //. c[n_] -> 1
Save["pl3.out", pl3, pl3point, pl3line, pl3total]
```

The coefficients in this generating function count the labeled graphs in each classification. If there is only one unlabeled graph with the given spectra, this number is that graph's multiplicity. More generally, in accordance with Proposition 1, the number of labeled graphs is

$$\sum_G \frac{p!}{\mathrm{Aut}(G)}\,, \tag{11}$$

where the sum is over all graphs with the given spectra. This identity provides a check that all graphs have been found and their automorphism groups correctly recorded.

2.5 Specific results

From the output of the *Mathematica* programs described above, tables of (p, q)-graphs up to $p + q = 8$ were constructed by hand as follows:

1. From the t^{2q} term in the generating function `n`p`line`, read off the possible line spectra and the total number of graphs corresponding to each. These spectra label the rows of the table.
2. From the t^{2q} term in the generating function `n`p`point`, read off the possible point spectra and the total number of graphs corresponding to each. These spectra label the columns of the table.
3. For each box in the table, record the number of unlabeled and of labeled graphs, from the generating functions `pu`p and `pl`p.
4. Fill in all the disconnected graphs. This is easily done recursively from previously constructed tables for smaller p, from which one considers only the connected graphs: For each k from 2 to p, consider all ways of partitioning the p points into k subsets; consider all ways of distributing the q lines among the k subsets; list all connected multigraphs on each subset with the given number of lines, and take the Cartesian product. For example, with 3 points and 2 lines (Figure 2),
   - if there is one connected subgraph of two points and one of one point, then there are
     - two graphs with 2 lines on the big subgraph,
     - one graph with 1 line on the big subgraph and 1 on the small one,
     - no graphs with both lines on the small one;
   - if there are three subgraphs of one point each, then there are
     - one graph with 2 lines on one subgraph,
     - one graph with 1 line on each of two subgraphs.
5. By subtraction, find the number of connected graphs for the box in question. Construct all such graphs. (This is easy because the number is small — just 1 in the example Figure 2. Adopting a canonical ordering for the points, as described in Sec. 3.1, helps to avoid listing isomorphic graphs by mistake.)
6. For each graph, note its automorphism group, whether it is connected, and whether it is simple. Check the automorphism numbers against the count of labeled graphs by (11).

The tables of graphs are presented pictorially in Appendix C and, more completely, in Fulling (1995).

Figure 3 presents a table of the total numbers of graphs of type (p, q), classified according to connectedness and simplicity to the extent that such statistics are known. Statistics for connected graphs appear at the

[Please see separate page.]

Figure 3: Counting general graphs. See Sec. 2.5 for explanation.

top of each box, those for disconnected graphs at the bottom. Statistics for simple graphs appear at the left, those for others at the right. Totals of adjacent statistics appear between them in larger type. Thus, for $(p, q) = (4, 4)$, the table means that there are 2 connected simple graphs, 7 connected nonsimple graphs (having either loops or multiple links), and 44 disconnected graphs (none of them simple); therefore, there are 2 simple graphs, 51 nonsimple graphs, 9 connected graphs, 44 disconnected graphs; a total of 53 graphs in all.

Similarly, the column heading for each p shows the numbers of simple graphs (connected, disconnnected, and total) on that number of points with arbitrarily many lines. (Numbers that go beyond the data in the body of the table come from Harary and Palmer (1973).) The row heading for each q shows the numbers (when available) of connected graphs (simple, nonsimple, and total) with that number of lines and arbitrarily many points.

The lower of the two heavy broken lines in Figure 3 is the top boundary of the region in which there are no simple graphs. The higher heavy broken line is the bottom boundary of the region in which there are no connected graphs. (See Prop. 2.)

2.6 General results

As indicated in Figure 3, certain patterns quickly become apparent in the enumerations. In particular, the situation for a given q eventually stabilizes for sufficiently large p, and can be stated once and for all in a general theorem.

Theorem 3

1. *When $q = 0$ and $p \geq 2$, there is one graph, which is simple and disconnected and has automorphism number $p!$.*
2. *When $q = 1$ and $p \geq 3$, there are two graphs, both disconnected, one simple with automorphism number $2(p-2)!$ and one nonsimple with automorphism number $(p-1)!$.*
3. *When $q = 2$ and $p \geq 4$, there are seven graphs, all disconnected, two simple with automorphism numbers $2(p-3)!$ and $4(p-4)!$, five others with automorphism numbers $(p-1)!$, $(p-2)!$, $2(p-2)!$, $2(p-2)!$, and $2(p-3)!$.*
4. *For fixed $q \geq 2$, for all $p \geq 2q$ the number of graphs in each classification (by connectedness, simplicity, and spectra) is independent of p. The graphs are obtained by adding isolated points to graphs with $p = 2q$.*

PROOF **2** *The maximum number of points that can be attached to q lines is $2q$. Any additional points must be isolated. Their addition does not change the simplicity or connectedness of the graph (except that connectedness can be lost when $q = 1$ and p increases from 2 to 3, and when $q = 0$ and p increases from 1 to 2). Nor does it change the line and point spectra, apart from the padding by trailing zeros. The specific results for $q = 0, 1, 2$ are obvious from inspection of the tables in Appendix C. (For $q = 2$, the only case missing from Fig. 2 is the graph with 4 points connected in pairs by 2 lines.)*

The asymptotic behavior in the vertical direction is more complicated, except for the first column, which is trivial:

Theorem 4 *When $p = 1$, there is only one graph, which is connected, is not simple except when $q = 0$, and has automorphism number 1.*

3 Preliminary Skirmishes

Along the way to the systematic attack described in the previous section, we developed some partial solutions to the problems of Sec. 1.3 that retain some independent interest.

3.1 A canonical ordering

There are two interrelated ordering issues: how to order the vertices within a given graph, and how to order different graphs within a list.

We have found it helpful to adopt the pictorial convention that the points in a labeled graph are always arranged in a line in numerical order, as in Figure 1. When the labeled graph is in canonical form, the corresponding diagram with unlabeled points is the canonical representation of the associated unlabeled graph. Although these diagrams may be less attractive than the more two-dimensional ones common in the literature of graph theory, they are free from the gratuitous extra automorphic ambiguity that comes from the freedom to move labeled points around the page.

We propose to order labeled graphs this way:

Definition 10 Regard the nonredundant elements of the adjacency matrix as the digits of a number, with respect to some sufficiently large radix, reading the digits in this order:

$$\begin{array}{cccc} . & 6 & 3 & 1 \\ & . & 5 & 2 \\ \nwarrow & & . & 4 \\ & & & 7 \end{array} .$$

Order the labeled graphs according to their numbers. (More formally, the coarsest ordering is in ascending order of the corner matrix element l_{1j}, the next coarsest in ascending order of l_{2j}, next $l_{1,j-1}$, next l_{3j}, next $l_{2,j-1}$, ...) The normal form of an unlabeled graph is the labeled instance of it that comes earliest in the ordering. □

The ordering hereby defined is not suitable for "orderly generation" in the sense of Read (1978). However, our primary purpose here has been to bring some conceptual order into the graph expansions of Theorem 1 and Corollary 1, and for this the proposed ordering has some aesthetic advantages (see below).

The function `Compositions[`q`, `$p(p+1)/2$`]` in the *Mathematica* package `Combinatorica.m` of Skiena (1990) generates a list of lists of integers; each list of integers is interpretable via Definition 10 as the nonredundant part of the adjacency matrix of a (general) (p, q)-graph, and these matrices arise in precisely the canonical order established in that definition. Appendix A contains a short *Mathematica* program to display this list of adjacency matrices for given (p, q). (Since the entire list is constructed at once, this program is not practical for large values of p and q.) The reader who tries it out will observe these properties and advantages of this canonical ordering:

1. Points connected by a great many links are preferentially placed as close together as possible. This is the big advantage of Definition 10 over the most obvious alternative, which is to order the points by the number of links attached to them (cf. operator Θ in Theorem 2). Simpler-looking drawings result. Disconnected graphs are automatically factored into components. In the physical application, when the terms are written out as tensor contractions, the summed index pairs are on adjacent factors as often as possible; thus, to the extent that a combination of tensors can be regarded as "matrix multiplication", that structure is made manifest.

2. Other things being equal, large numbers of links are moved as far to the left as possible, followed at a lower level of priority by large numbers of loops.

3. In the resulting ordered list of unordered graphs, graphs of a similar structure tend to be grouped together. In particular, disconnected graphs tend to come before connected ones. This separation is not complete, however: When $p = 4$ and $q = 3$,

$$\begin{array}{cccc} 0 & 1 & 0 & 0 \\ & 0 & 1 & 0 \\ & & 0 & 1 \\ & & & 0 \end{array} \quad \text{(connected)} \qquad \text{precedes} \qquad \begin{array}{cccc} 0 & 1 & 1 & 0 \\ & 0 & 1 & 0 \\ & & 0 & 0 \\ & & & 0 \end{array} \quad \text{(disconnected).}$$

Although we have not employed this ordering of *all* (p, q) graphs elsewhere in this paper, in the charts we have ordered the graphs *in each box* and the points in each graph in the way described here.

3.2 Permuting points

Appendix B contains some *Mathematica* functions useful in studying the effect on an adjacency matrix of permuting the points of the graph.

3.3 Line spectrum: Enumeration by Polya's theorem

The generating function $N(1, \ldots, 1)$ for general graphs on p points, enumerated by line spectrum, can be constructed directly, as an instance of *Pólya's theorem* (Harary and Palmer, 1973, esp. pp. 84ff). The procedure for applying Pólya's theorem is rather standard, without some of the complications of Parthasarathy's construction.

One looks at the action of the permutation group S_p upon the $\frac{1}{2}p(p+1)$ possible line positions, and decomposes each group element into cycles. A variable x_k is associated with each k-cycle. For example, when $p = 3$ there is one permutation (the identity) of type ${x_1}^6$, three (the transpositions) of type ${x_1}^2{x_2}^2$, and two (the cyclics) of type ${x_3}^2$. This analysis (cf. (7)–(10)) leads to the *cycle index*

$$P_6 = \frac{1}{6}({x_1}^6 + 3{x_1}^2{x_2}^2 + 2{x_3}^2).$$

Then one substitutes the formal series $\sum_i C_i^k$ for x_k :

$$P_6 = \frac{1}{6}\left[\left(\sum C_i\right)^6 + 3\left(\sum C_i\right)^2\left(\sum C_i^2\right)^2 + 2\left(\sum C_i^3\right)^2\right].$$

To count the possible arrangements of $q = 5$ (for example) lines, one extracts from this sum the terms of the form

$$C_0^{j_0} C_1^{j_1} \cdots C_5^{j_5} \quad \text{with } j_1 + 2j_2 + \cdots + 5j_5 = 5$$

(in addition to the automatic $j_0 + \cdots + j_5 = 6$). The coefficient of $C_0^2 C_1^3 C_2$ is the number of multigraphs with line spectrum 2111, and so on.

3.4 Point spectrum: Enumeration by invariant theory

It is interesting to look at the enumeration of general graphs by point spectrum from a different point of view. In the context of the application described in Sec. 1.2, simplified by setting $\mathbf{y}$ (and hence all the $\mathbf{z}_i$) equal to $\mathbf{x}$, a (p, q)-graph with point spectrum $k_1 k_2 \cdots k_p$ stands for a product $\nabla^{k_1} V \cdots \nabla^{k_p} V$, where the $2q = \sum k_i$ variables of differentiation (components of $\mathbf{x}$) are set equal and summed in pairs. The enumeration problem thus is the problem of finding all possible such pairings (taking into account the commutativity of factors V and the symmetry of repeated partial differentiation).

This is an easy special case of a general problem of tensor calculus, that of finding all polynomial invariants of a given degree that can be constructed from a given set of tensor fields by contraction of index pairs, taking into account the index symmetries of the tensors. The technology for doing this for the Riemann curvature tensor and its derivatives has recently been treated by Fulling, King, Wybourne, and Cummins (1992). The present case of a scalar potential is significantly easier, because all of the tensors $\nabla^k V$ are totally symmetric. It requires the calculation of outer products and plethysms of irreducible representations of the S_{k_i} labeled by Young diagrams of only one row. Such calculations are best done with the aid of the program SCHUR (Wybourne, 1992), and, of course, they agree with $N(t, \ldots, t)$ found by the method of this paper.

A Appendix

The *Mathematica* function `adjMat` converts a list of numbers to a symmetric matrix in the way prescribed in Definition 10. The function `labGraphs` produces the entire list of (p, q)-graph adjacency matrices and formats them as matrices (rather than lists of lists) for output.

```
Needs["DiscreteMath`Combinatorica`"]

adjMat[l_List, p_] := Block[{m, jj, i, r, c, hold},
   jj=p(p+1)/2;
   r=1;
   c=p;
   Do[
      m[r,c] = l[[i]];
      m[c,r] = m[r,c];
      If[r==1,
         hold=c; c=p; r=p-hold+2,
         r=r-1; c=c-1],
    {i, jj}];
   Array[m, {p,p}]]

labGraphs[p_, q_] := Map[MatrixForm[adjMat[#,p]]&,
                                        Compositions[q, p(p+1)/2]]
```

B Appendix

The *Mathematica* function `basisPerm` performs on an adjacency matrix the transformation induced by a permutation `f` of the points of the labeled graph. The transposition of the `a`th and `b`th points is constructed as `transposition[a,b]` and implemented on the adjacency matrices by the function `tr`; other permutations could be constructed similarly.

```
transposition[a_,b_,a_] := b
transposition[a_,b_,b_] := a
transposition[a_,b_,c_] := c
transposition[a_,b_] := transposition[a,b,#]&
listPerm[f_,l_List] := Array[l[[f[#]]]&, Length[l]]
rowPerm[f_,m_] := listPerm[f,m] /; MatrixQ[m]
columnPerm[f_,m_] := Map[listPerm[f,#]&, m] /; MatrixQ[m]
basisPerm[f_,m_] := MatrixForm[rowPerm[f,columnPerm[f,m]]]
tr[a_,b_,m_] := basisPerm[transposition[a,b], m]
```

C Appendix

Here are the tables listing all unlabeled (p, q)-graphs with $p + q = 8$, classified by line spectrum (rows) and point spectrum (columns). The full set of tables for $1 \leq p + q \leq 8$ is available on the World Wide Web (Fulling, 1995), along with additional information relating to this project.

[**Please see separate pages.**]

Acknowledgement: The computations were done on a Sun 3/60 workstation with *Mathematica* software, provided by a donation from Sun Microsystems extended by Wolfram Research and instigated by S. M. Christensen. We thank B. G. Wybourne and A. Kerber for drawing our attention to MOLGEN and some relevant literature, and an anonymous referee of an earlier version for corrections and improvements.

References

[1] I. G. Avramidi, A covariant technique for the calculation of the one-loop effective action, *Nucl. Phys. B* **355** (1991) 712–754.

[2] A. O. Barvinsky, T. A. Osborn, and Yu. V. Gusev, A phase-space technique for the perturbation expansion of Schrödinger propagators, *J. Math. Phys.* **36** (1995) 30–61.

[3] C. Benecke, R. Grund, R. Hohberger, A. Kerber, R. Laue, and T. Wieland, Chemical isomerism, a challenge for algebraic combinatorics and for computer science, in: G. Cohen, M. Giusti, and T. Mora, eds., *Applied Algebra, Algebraic Algorithms and Error-Correcting Codes* (*Lecture Notes in Computer Science* **948**) (Springer, Berlin, 1995) 4–20.

[4] Y. Fujiwara, T. A. Osborn, and S. F. J. Wilk, Wigner–Kirkwood expansions, *Phys. Rev. A* **25** (1982) 14–34.

[5] S. A. Fulling, Cataloguing general graphs by point and line spectra: The World Wide Web site, URL `http://www.math.tamu.edu/~fulling/graphs/` (1995).

[6] S. A. Fulling, R. C. King, B. G. Wybourne, and C. J. Cummins, Normal forms for tensor polynomials: I, *Class. Quantum Grav.* **9** (1992) 1151–1197.

[7] P. B. Gilkey, The spectral geometry of a Riemannian manifold, *J. Diff. Geom.* **10** (1975) 601–618.

[8] L. A. Goldberg, *Efficient Algorithms for Listing Combinatorial Structures* (Cambridge University Press, Cambridge, 1993).

[9] F. Harary and E. Palmer, *Graphical Enumeration* (Academic Press, New York, 1973).

[10] A. Kerber et al., MOLGEN — Automatic structure elucidation in chemistry, URL `http://www.mathe2.uni-bayreuth.de/molgen/mghome.html` (1996).

[11] L. Kučera, *Combinatorial Algorithms* (Adam Hilger, Bristol, 1990).

[12] F. Molzahn, Exponential cluster solutions to quantum transport equations, *J. Phys. A* **25** (1992) 4913–4940; erratum *ibid.* **26** (1993) 2275–2276.

[13] F. H. Molzahn and T. A. Osborn, Connected graph representations of the quantum propagator and semiclassical expansions, *J. Phys. A* **20** (1987) 3073–3094.

[14] F. H. Molzahn and T. A. Osborn, A phase space fluctuation method for quantum dynamics, *Ann. Phys. (N.Y.)* **230** (1994) 343–394.

[15] T. A. Osborn and F. H. Molzahn, The Wigner–Weyl transform on tori and connected graph propagator representations, in: S. A. Fulling and F. J. Narcowich, eds., *Forty More Years of Ramifications: Spectral Asymptotics and Its Applications* (*Discourses in Mathematics and Its Applications* **1**) (Mathematics Department, Texas A&M University, College Station, TX, 1991), 199–236.

[16] T. A. Osborn and F. H. Molzahn, Moyal quantum mechanics: The semiclassical Heisenberg dynamics, *Ann. Phys. (N.Y.)* **241** (1995) 79–127.

[17] K. R. Parthasarathy, Enumeration of ordinary graphs with given partition, *Canadian J. Math.* **20** (1968) 40–47.

[18] G. Pólya and R. C. Read, *Combinatorial Enumeration of Groups, Graphs, and Chemical Compounds* (Springer, New York, 1987).

[19] R. C. Read, Every one a winner, in: B. Alspach, P. Hell, and D. J. Miller, eds., *Algorithmic Aspects of Combinatorics* (*Annals of Discrete Mathematics* **2**) (North-Holland, Amsterdam, 1978) 107–120.

[20] R. C. Read and D. G. Corneil, The graph isomorphism disease, *J. Graph Theory* **1** (1977) 339–363.

[21] S. Skiena, *Implementing Discrete Mathematics: Combinatorics and Graph Theory with Mathematica* (Addison–Wesley, Redwood City, CA, 1990).

[22] T. Wieland, A. Kerber, and R. Laue, Principles of the generation of constitutional and configurational isomers, *J. Chem. Inf. Comput. Sci.* **36** (1996) 413–419.

[23] B. G. Wybourne, *SCHUR: An interactive programme for calculating properties of Lie groups* (1992).